OIPE JC1&3
OCT 01 2001
PATENT & TRADEMARK OFFICE

RECEIVED
OCT -3 2001
TC 2600 MAIL ROOM

(11) Laid-Open Patent Application Publication No. Hei-6-308958
(12) Laid-Open Patent Application Publication (A)
(43) Publication Date: November 4, 1994
(51) Int Cl[5] : G10H 1/02, 1/043, 1/057

Request for Examination: not yet requested
Number of Claims: 1

(22) Application Date: April 26, 1993
(21) Application No. Hei-5-99515
(71) Applicant: Roland Kabushiki Kaisha (Code: 000116068)
1-4-16, Dojimahama, Suminoe-ku, Osaka, Japan
(72) Inventors: Norio Matsushika, Shinobu Katayama, and Tsuyoshi Sakata
c/o 3-7-13, Shinkitashima, Suminoe-ku, Osaka, Japan

(54) Musical Tone Control Device for Electronic Musical Instrument

[Detailed Description of the Invention]

[0001]

[Field of the invention]

The present invention relates to a musical tone control device which is incorporated in an electronic musical instrument and provides sequential change in the state of a musical tone being under generation, that is, the change of characteristics, such as tone pitch, tone quality and tone volume of the musical tone being under generation.

[0002]

[Related prior art]

It is well known that an electronic musical instrument is allowed to perform more expressively by provision of sequential change in characteristics (such as tone pitch, tone quality and tone volume) of musical tones being under generation. Conventionally, such tone control devices sequentially changing tone characteristics have been incorporated in electronic musical instruments. Generally, in an electronic musical instrument a plurality of tone generating channels (hereinafter referred to as "voice") are provided, each channel being in charge of generating one musical tone at a

time. To change characteristics of a musical tone being under generation, the state transition (variation in characteristics of the musical tone) for each voice is computed respectively, every predetermined period in the manner of time-sharing and then, according to the result of the computation each voice is controlled. This controlling is herein referred to as "voice service" (or merely as "service"). Such voice service is typically implemented with CPU.

[0003]

[Problems to be solved by the invention]

Incidentally, the capability of generating multiple musical tones is preferable for colorful musical performance. To attain this advantage the increased number of voices are provided, resulting in the increase of the voices to be serviced. Basically, the service for each voice is carried out at each predetermined period. However, in addition to the voice service the CPU executes usually many other processes in the manner of time-sharing. In this case, the CPU could not service every predetermined period if the load other than the service, such as MIDI signal receiving process, becomes temporarily heavy. In such a case, there occurs a problem such that the period of LFO (Low Frequency Oscillator) may vary when generating LFO waveform as a service content, or the profile of generated envelope waveform may become long relative to the time axis when generating envelope waveform.

[0004]

Though the elongation of the period can be adopted to overcome the problem described above, the change of musical tone characteristics becomes coarse to cause another problem of the deterioration in tone quality. To prevent the deterioration in tone quality, while keeping the service period short and allowing the increased number of voices to be serviced, a high-speed CPU may be used, but it causes the cost to be increased.

[0005]

In view of the circumstances described, the object of the invention is to provide a musical tone control device of an electronic musical instrument for servicing adequately to voices even if the service period is varied.

[0006]

[Means for solving the problem]

To achieve the object, the musical tone control device for an electronic musical instrument according to the invention, comprising:

(1) a musical tone generating means for generating a musical tone and for changing the characteristics of the musical tone being under voicing according to musical tone

control data supplied;

(2) a control data creating means for creating said musical tone control data every predetermined period and for supplying said created control data to said musical tone generating means; and,

(3) a control-data-value varying means for controlling variably the value of the musical tone control data created in said control data creating means, according to a variation of the actual period relative to the predetermined period to be serviced.

[0007]

[Action]

In the musical tone control device of the invention, if the actual period computed varies relative to the predetermined period to be serviced, then values of musical tone control data can be variably controlled on the basis of the variation. The variation of the service period is therefore allowed.

[0008]

[Example]

Now, an example of the present invention will be described below. First of all, is discussed the case such that first service is needed:

(1) When the envelope is steep:

FIG. 8 shows a typical envelope of a musical tone.

[0009]

In the figure, the envelope changes such that the envelope rises steeply just after keying (attack), then lowers a little (decay), holds on the musical tone (sustain), and lowers quickly after releasing the key (release). For the envelope in this drawing, the inclination during the decay D is larger than that during the sustain S, and the inclinations during the attack A and release R are furthermore larger than others. Accordingly, it is necessary to service in short period during the attack A and release R, meanwhile, the service can be reduced efficiently during the sustain S.

[0010]

(2) When LFO rate is large, or LFO depth is large:

When vibrato or tremolo effects by LFO (Low Frequency Oscillator) is large, in other words, the change of tone pitch or tone volume per constant time is large (*i.e.*, in deep modulation, in fast modulation), it is necessary to service steadily in short period. Meanwhile, when the modulation is shallow and slow, the service can be reduced efficiently.

[0011]

(3) Just after the CC (Operator) is operated;

Just after the CC (continuous controller) such as a bender lever or level slider is operated, or just after the equivalent MIDI data (performance data) is received, it is necessary to service immediately. In an example to be discussed below, the service period denominated in this invention which supplies tone control data to the tone generating means is obtained according to the changing quantity of tone characteristics per predetermined time which is generated in accordance with performance data. That is, in the example described below, the three conditions (1) through (3) described above are observed at each voice, the shortest service period among these three service periods computed from (1) through (3), respectively, is then taken as the service period for the voice at that point of time. By carrying out the voice service in the service period obtained in such a way, the voice service is thus carried out in short period when the variation of the tone characteristics is large, meanwhile in long period when the variation is small. Because the voice service period is fixed, it has hitherto been necessary that the service period be set for the big change of the musical tone characteristics. But, it is exceptional that the big change of the musical tone characteristics which are currently generated for many voices occurs simultaneously. In this example, as described above, the period is made long when the change of the musical tone characteristics is small, so that the musical tone with big change of the musical tone characteristics can be serviced by the margin of capacity. Thus, such constitution using relatively low-speed processing means can implement the service for a number of voices without any deterioration of the tone quality.

[0012]

Once the service period has been obtained as described above, the musical tone control data in this example is generated every period obtained as above, according to the time point of the currently generated musical tone., and the generated musical tone control data is supplied to the musical tone generating means so-called in this invention. In this example, since the constitution can change the musical tone characteristics with the musical tone control data generated according to the time point of the musical tone being under generation, the tone control can be executed precisely even though a number of tone characteristics exceptionally vary enormously and simultaneously, or the processing is omitted (*i.e.*, the service period becomes long.) or delayed as a result of such a case that or the time to be serviced passes away because CPU executes another process.

[0013]

FIG. 1 is a block diagram of the construction of an electronic musical instrument in which an example of the musical tone control device of this invention is incorporated.

This electronic musical instrument generates and outputs the tone signal according to a received MIDI signal (performance data). The control section 1 comprises a CPU for executing various programs, a ROM for storing programs and various data for the execution of program or the like, a RAM used for working area of the execution of program, a timer for generating interrupt signal while measuring time, and others. In the control section 1, the control data is created according to MIDI data, and the control data is then supplied to the musical tone generating circuit 2. Though the electronic musical instrument in the example receives MIDI signal and then generates musical tone signal according to the MIDI signal, it will be understood that the electronic musical instrument has keyboard or operators for performance and generates tone signal according to the performance data created by prepared keyboard operation, instead of receiving MIDI signal or along with MIDI signal.

[0014]

The tone generating circuit 2 is constituted by a plurality of hardwares, each (herein referred to as "voice") generating single musical tone signal so that a plurality of tone can be generated simultaneously. The control section 1 selects a vacant voice when Note On signal which is one of the MIDI signal is inputted; sends the voice the control data such as tone pitch, tone quality, tone volume, or the like; and allows the voice to generate tone signal. When Note Off signal is inputted to the voice which currently generates the musical tone and corresponds to the Note Off signal, the control section 1 supplies the control data for stopping the musical tone generation. The control section 1 furthermore computes periodically the envelope signal and LFO (Low Frequency Oscillator) signal of each voice which currently generates musical tone, then supplies the control data for controlling each parameter of tone pitch, tone quality (such as cutoff frequency of filter), tone volume or the like according to the operational signal of the performance operator (such as bender lever, modulation wheel and expression pedal, *etc.*) given through MIDI signal. The present invention relates to the process for providing control data periodically.

[0015]

The entire program which is executed by CPU in the control section 1 is constituted with the type of multiple task. The starting condition of each task is monitored by monitor program (hereinafter referred to as "monitor"), and if the starting condition is met, each task corresponding to the starting condition is started. Only the voice processing task (which supplies the control data, such as tone pitch, tone quality and tone volume or the like, to each voice according to the signal of envelope, LFO and operator) related to the present invention is shown herein.

Incidentally, the start or stop process for each musical tone generation is executed by the musical tone generating start or stop task (not shown) which is started under the condition of receiving Note On or Note Off signal.

[0016]

The task of voice processing executed by CPU in the control section 1 is now described below. For the simplicity, the same reference symbols may be used between each register or flag and the contents stored in each register or flag without distinction. FIG. 2 is a flowchart showing the main routine of the task of voice processing. The task of voice processing is started by the monitor as follows. That is, the program execution time is measured by the interrupt routine (not shown) started by interruption with timer, and the monitor attempts to start the task of voice processing at each constant time (the period of starting the task of voice processing, for example, 8 ms) based on the measuring result. At this time, if the task which has a higher priority (*e.g.*, the task of MIDI signal receiving process, or the task of starting/stopping musical tone generating, *etc.*) than the task of voice processing, or the task of voice processing itself is still executed, the task of voice processing is started after finishing the process of the task. Accordingly, though the task of voice processing should be basically started at constant time, the starting could be late according to the circumstances of processing the task in CPU at the time of attempting to start.

[0017]

This task of voice processing turns in the state of waiting START by the monitor (referred to as "TIMER WAITING") after sequential process (Step (2_1)). Proceeding to Step(2_2) after starting by the monitor, the value added 1 to "NUMBER OF TIMER OVERFLOW TIMES" is stored in the register DELAY. "NUMBER OF TIMER OVERFLOW" is added 1 by the monitor every predetermined period of the task of voice processing if the monitor could not start the task of voice processing because of existing another executing task when the monitor attempts to start this task of voice processing. This NUMBER OF TIMER OVERFLOW TIMES is cleared to "0" by the monitor when the value is read out by the task of voice processing. That is, NUMBER OF TIMER OVERFLOW shows how the process of the task of voice processing is delayed. This value is "0" when the task of voice processing can start immediately because of without executing task.

[0018]

The number of times that the starting was delayed (*i.e.*, NUMBER OF TIMER OVERFLOW) plus one is stored in DELAY such as "1" when this task of voice processing is started immediately, "2" when the task has been delayed once. The

subsequent processes (Step (2_3) through Step (2_7)) of the routine of voice processing create the control data and provide it to each currently generating voice respectively. Incidentally, the CPU load is intended to reduce by omitting the process for the voice which is not currently executing the musical tone generating process (*i.e.*, vacant voice).

[0019]

The subroutine of voice processing in Step (2_5) shown in FIG.2 will be then described. In this subroutine, although the same process for each parameter of tone pitch, tone timbre and tone volume is independently and respectively executed, only one representative parameter is shown and described for the simplicity. FIG. 3 is a flowchart of a subroutine of voice processing.

[0020]

When this subroutine of voice processing is started, first, in Step (3_1), the value of register NEXT_SRV (VOICE) which is prepared respectively corresponding to each voice (voice No. VOICE) is reduced by DELAY. This NEXT_SRV (VOICE) shows scheduled elapse time in units of the starting period of the task of voice processing. The scheduled elapse time is the service period obtained previously by subroutine NEXT_SRV (VOICE) at Step (3_4) described later, that is, the time until the process for supplying control data will be executed for this voice at next time. In other words, when the starting period of the task of voice processing is 8 ms, NEXT_SRV (VOICE) = 1 signifies 8 ms. In the process of Step (3_1), NEXT_SRV (VOICE) shows the residual time of the scheduled elapse time from previous service to current service by reducing DELAY from the value of NEXT_SRV (VOICE). That the value of NEXT_SRV (VOICE) becomes zero or negative means the time to be newly serviced for the voice. (refer to Step (3_3)).

[0021]

Then, in Step(3_2), whether the operator is newly operated or not after the previous process is estimated. The operator denotes herein the operation reflected to the parameter which is the object of process in this subroutine of voice processing. The case, for example, that the process for tone pitch parameter is executed in this subroutine of voice processing will be discussed. When the tone pitch is set as being controlled by bender lever, it can be determined that the new operation has been done if the bender lever was operated. Meanwhile, when the tone pitch is not set as being controlled by bender lever, it can be determined that the new operation has not been done even though the bender lever was operated. When the variation of parameter for the operational movement of operator, that is, the sensitivity of operator is also

changeable, it can be determined that the operator was operated if the sensitivity was changed.

[0022]

The reason of distinguishing the processes depending on whether the operator is newly operated or not is that the service has to be done immediately before the predetermined service timing if the operator was operated. First, the case that the new operation by operator was not done will be described below. If it is determined that new operation is not done in Step (3_2), the Step (3_3) will be executed. In Step (3_3), it is determined whether NEXT SRV (VOICE)≤0 or not, that is, whether the new service timing comes or not, for the currently processing voice. When the determined result is indicative of NEXT SRV (VOICE)>0, that is, the new service timing has not come yet, the subroutine of voice processing is ended. Meanwhile, when the determined result is indicative of that the new service timing has come, the process proceeds to Step (3_4). In Step (3_4), the new value of NEXT SRV (VOICE), that is, the service interval from present time point to the time to be serviced next, in relation to this voice is obtained. The execution subroutine of NEXT SRV (VOICE) in this Step (3_4) will be described later. After the new value of NEXT SRV (VOICE) is obtained by Step (3_4), the process progresses to Step (3_5), and the value of COUNT (VOICE) indicative of lapsed time (the position on the horizontal axis in FIG. 8) from the initial time of voice generation (corresponding to "keying" position in FIG.8) is obtained. Incidentally, COUNT (VOICE) is cleared to "0" when the voice generation is started in the task of starting tone generation (not shown). Since the value of COUNT (VOICE) before being changed in Step (3_5) is indicative of the lapsed time at the previous service time, SRV_OLD (VOICE)+DELAY-1 which is the lapsed time from previous service time to the present time is herein added to the lapsed time COUNT at the previous service time. SRV_OLD (VOICE) is the predetermined lapsed time from the previous service to the current service, that is, NEXT_SRV (VOICE) obtained at the previous service (refer to Step (3_7)), and DELAY-1 is the process time delayed, that is, NUMBER OF TIMER OVERFLOW described above. The current lapsed time from the time point corresponding to keying, that is, the present position on the horizontal axis in FIG. 8 by computing in Step (3_5).

[0023]

In addition, the method of computing process other than the method of computing process described above may be adopted in this Step (3_5), and in Step (3_8) described later. In other word, for example, the method of computing process may be adopted the method such as obtaining directly the time from the starting time by

measuring the time in timer interrupt routine, or adding the lapsed time from the starting of tone through the previous service to the lapsed time obtained from previous service time. Alternatively, the position in single period of LFO waveform, or the position in the phase of envelope of waveform (attack, decay, sustain or release *etc.*) may be used rather than the lapsed time from the voice starting of tone generation. In other words, the sequential position at service point has only to be obtained even though the delayed process exists.

[0024]

Once the new value of NEXT_SRV (VOICE) and COUNT (VOICE) have been obtained, in proceeding Step (3_6), the control value to be attained at next service time (after NEXT_SRV (VOICE)) is obtained and provided to the voice. This detail of voice control subroutine (Step (3_6)) will be described later. After the control data is provided to the voice, SRV_OLD (VOICE) is updated (Step (3_7)).

[0025]

When it is determined that the operator has been newly operated in Step (3_2), the value of new COUNT, that is, the present position, is obtained on the horizontal axis in FIG. 8 in the proceeding process Step (3_8). As described above, SRV_OLD (VOICE) is the predetermined lapsed time of the present service accounted from the time point of previous service, and NEXT_SRV (VOICE)) is indicative of the residual time of this predetermined lapsed time because of being reduced sequentially in Step (3_1). Accordingly, SRV_OLD (VOICE) - NEXT_SRV (VOICE)) is indicative of the real lapsed time from the previous service to the present service. Hence, the time point of the present service is obtained by adding SRV_OLD (VOICE) - NEXT_SRV (VOICE)) to the value of COUNT before updating which represents the time point of the previous service.

[0026]

"1" is stored in the NEXT_SRV (VOICE) in the proceeding Step (3_9) after Step (3_8). This is for setting the new service interval in the next service timing according to the operated operator (Step (3_4). In the example, as described later, the service interval is determined according to the rate of the LFO waveform or envelope waveform, but the service interval which is appropriate to the new rate comes to be set immediately even though the rate of the LFO waveform or envelope waveform is changed by the operated operator by executing the process described above when the operator is moved.

[0027]

After "1" is stored in NEXT_SRV (VOICE)) and obtained COUNT (VOICE) as

described above, the process proceeds Step (3_6), and the control value which is to be attained at next service time (NEXT_SRV (VOICE))=1) is obtained. FIG. 4 shows the flowchart of NEXT_SRV (VOICE)) computing subroutine which is shown in Step (3_4) in FIG. 3. The service interval to the next service is obtained herein.

[0028]

In this subroutine, LFO_NEXT (VOICE) and EG_NEXT (VOICE) are obtained in each Step (4_1) and (4_2) respectively, and the smaller one in the two values is set as NEXT_SRV (VOICE)) in Step (4_3). LFO_NEXT (VOICE)) is the service interval obtained from the result of the multiplication of the rate of LFO waveform and the level related to the parameter. The larger the rate is, and the higher the level is, the smaller value is set in this LFO_NEXT (VOICE)), *i.e.,* this signifies that the service interval is shortened. The service interval, for example, is set to "1" when the product of multiplication is larger than the predetermined value, and is set larger according to such as the reciprocal number of the portion when that is smaller. For example, when the portion is 1/2, the value of service interval LFO_NEXT (VOICE)) is set to "2". In other words, the tone control is precisely done by shortening the service interval when the variation quantity during the predetermined time of the value of the LFO waveform, and adversely the load of CPU is reduced by elongating the service interval when the variation is small.

[0029]

EG_NEXT (VOICE)) is the service interval obtained from the rate (*i.e.,* gradient) of the envelope waveform related to the parameter (EG stands for "Envelope Generator"). The larger the rate is, the smaller value is set in this EG_NEXT (VOICE)). That is, the service interval is shortened. Even in this case, the service interval is set to "1" when the rate is larger than the predetermined value, and the service interval is elongated according to, for example, the reciprocal of the portion when the rate is smaller. In other words, the tone control is precisely done by shortening the service interval when the variation quantity during the predetermined time of the value of the envelope waveform, and adversely the load of CPU is reduced by elongating the service interval when the variation is small.

[0030]

In Step (4_3), by setting the smaller one of either LFO_NEXT (VOICE) or EG_NEXT (VOICE)) as NEXT_SRV (VOICE)), the service interval is determined according to the element having the largest variation during the predetermined time of the value of problematic parameter. Incidentally, the value of LFO_NEXT (VOICE)) has not to be computed every time. It can be carried out and stored only at the time

when either the rate or level changes. Correspondingly, the value of EG_NEXT (VOICE)) has also not to be computed every time. It can be carried out once and stored only at the time when the rate of envelope changes (*e.g.,* the envelope of waveform changes from attack A to decay D *etc.*). The load of CPU may decrease significantly because of the computing quantity in such a case can be reduced.

[0031]

Further, LFO_NEXT (VOICE)) or EG_NEXT (VOICE)) can be computed preliminarily according to the rate of the LFO waveform or the envelope waveform respectively and stored for a moment rather computing them in real time, and then can be read out for using at the necessary timing. For example, EG_NEXT (VOICE)), in the case of the envelope waveform, corresponding to the rate of the attack A or decay D respectively can be stored for the attack A or decay D respectively similarly to the rate.

[0032]

Although, in this example, it is not considered whether the delayed process exists or not on the occasion of obtaining the service interval, the service interval can, alternatively, be shortened than the service interval obtained above when the delayed process does not exist. The reason is that it signifies that the load of CPU has some margin when the delayed process does not exist and that the number of the currently generating voices is a few. In this case, the deterioration of the tone quality is highly audible.

[0033]

FIG. 5 shows the flowchart of the voice control subroutine shown in Step (3_6) in FIG. 3. In Step (5_1) or (5_2), the value of LFO waveform or envelope waveform can be obtained respectively at the preceding time indicated by NEXT_SRV (VOICE), that is, at the time indicated by the value which is added NEXT_SRV (VOICE) to COUNT shown the lapsed time from the starting of tone of the voice. Prior art may be utilized for the generating process of LFO waveform or envelope waveform.

[0034]

The following process can be used as an example. When the envelope waveform is generated, typically, the value indicating the rate (*i.e.*, gradient) of envelope at each predetermined time is accumulated, and the accumulating value is used as the envelope waveform. When the present invention is applied to such a method of process, the value indicating the rate of envelope is added to the current value of envelope waveform similarly to the conventional method if NEXT_SRV (VOICE) is "1", and two times of the value indicating the rate of envelope is added to the current value of envelope waveform if NEXT_SRV (VOICE) is "2". That is, the product of multiplication

of the value indicating the rate of envelope by the value of NEXT_SRV (VOICE) is added to the current value of envelope waveform. Incidentally, the similar method of process can be used for LFO waveform.

[0035]

In addition, in this example, although the value is obtained at the preceding time indicated by NEXT_SRV (VOICE), it is merely signified that the LFO waveform or envelope waveform is slide by the value which is indicated by NEXT_SRV (VOICE). Accordingly, it is not necessary to arrange as such described above. The value corresponding to the present time indicated by the value of COUNT may be obtained alternatively.

[0036]

Moreover, the value related to operation is obtained in Step (5_3), and the overall value of preceding control data by a factor of the time being indicative of NEXT_SRV (VOICE) according to the value of LFO waveform and envelope waveform and the value related to the operator is obtained in Step (5_4), and then these values are provided to voice. In other words, in this invention, since the interval providing control data to voice is elongated or shortened, the value of control data provided according to that is also corresponding to the interval.

[0037]

FIGs. 6 and 7 show the variation of the value of the control data provided to voice. The service interval is basically "3" in FIG. 6 and 7, FIG. 6 shows when the delayed process exists, and FIG. 7 shows when the operated operation exists. In each drawing, the time points denoted Δ on the horizontal axis show the time points serviced actually, and the values denoted X show the values of control data provided to voice at each time point. Additionally, a chain line shows the case that the tone characteristics changes ideally.

[0038]

As shown in FIG. 6, when the delayed process exists, the value of control data of the next service is compensated with the delayed time. Meanwhile, when the operator is operated, the service is done immediately. Additionally, in this invention, it was described that the tone generating circuit generates the tone signal of tone pitch, tone timbre and tone volume as indicated by the control data provided. However, since the interval provided control data is only a few ms when typical CPU is used, the tone pitch *etc.* could only change coarsely and Stepwise as shown by solid line in FIG. 6 and 7. To prevent this, the tone generating circuit 2 (refer to FIG. 1) may be constituted with a interpolation circuit so that the tone pitch *etc.* could change smoothly. In this

case, the following process may be carried out. That is, in addition to that the preceding control data by a factor of the time indicated by the service interval (NEXT_SRV (VOICE)) is provided from the control section 1 to the tone generating circuit 2 (*i.e.,* voice), the value of service interval (the value of NEXT_SRV (VOICE)) is also provided. The interpolation is carried out in the tone generating circuit 2 so that the tone pitch, tone timbre and tone volume become to those which is indicated by the control data provided after the time indicated by the service interval. This interpolation is carried out according to the value of service interval which is different in each voice. As the method of interpolation, the method of the process used for generating the envelope waveform, or the similar method as the method of process of tone pitch changing as portamento or the like can be adopted. For example, the integral circuit with a time constant corresponding to the time indicated by the value of the service interval. In this case, the tone pitch *etc.* approximates to the value indicated by control data drawing the exponential curve. Otherwise, the quotient (differential tone pitch *etc.*) of, the difference between the current tone pitch *etc.* and the tone pitch *etc.* indicated by the control data, divided by, the quotient of the value of service interval divided by the interpolation-computing period, is obtained so that the tone pitch *etc.* can be changed by this differential tone pitch at each interpolation-computing period. In this case, the tone pitch *etc.* approximates to those which is indicated by control data, drawing a straight line as shown by dotted line in FIG. 6 and 7. In addition, when the tone pitch *etc.* attained to those which is indicated by control data, the attained tone pitch *etc.* is held on.

[0039]

In addition, in the case of interpolating in the tone generating circuit 2, the tone pitch *etc.* of music which is generated in the tone generating circuit 2 can be changed smoothly compared to the change of the value of control data. Accordingly, the change is not directly reflected to the tone pitch *etc.* generated in the tone generating circuit 2 even though the value of control data changed. With this view, in the case that the service interval is set not to change despite of being operated the operator, when the interpolation is carried out according to the long service interval at the small rates of LFO waveform and envelope waveform, since the interpolation is carried out with the small interpolation rate, the change is not reflected to tone pitch *etc.* despite of having changed the value of control data by operating the operator. It results in the feeling of strangeness for a player. In this example, when the operator is operated, since the service interval is compulsorily set to be the shortest "1", the operation of the operator is reflected immediately to tone pitch *etc.* even though the service interval is

preliminarily set to be long so that the feeling of strangeness for player could not occur.

[0040]

In the example, although it has been described as the voice processing is collectively carried out according to smaller one of the values of service interval obtained for LFO waveform and envelope waveform respectively, the process of each waveform can be, alternatively, carried out independently according to the most suitable service interval for each waveform. Furthermore, the present invention may be applied to any control waveform providing sequential change to musical tone characteristics, as well as LFO waveform and envelope waveform.

[0041]

[Effect of the invention]

As described above, by the present invention, since the period of supplying control data to the musical tone generating means according to the variation of tone characteristics during the predetermined time is changed, the load of CPU which may be conventional can be reduced without the deterioration of tone quality. Accordingly, the more voices, compared with conventional manners, can be controlled. Further, the present time point under the processing is measured from the starting point of tone, and the control data is obtained according to the present time point, so that the precise tone control could be carried out regardless that the change of processing period exists or the delayed processing exists or not.

[Brief description of the drawings]

FIG. 1 is a block diagram of the construction of an electronic musical instrument in which an example of the musical tone control device of this invention is incorporated.

FIG. 2 is a flowchart, showing a main routine of the task of voice processing.

FIG. 3 is a flowchart of a subroutine of voice processing.

FIG. 4 is a flowchart of a subroutine of NEXT_SRV(VOICE) computation.

FIG. 5 is a flowchart of a subroutine of voice control.

FIG. 6 shows the variation of the control data value supplied to the voice.

FIG. 7 shows the variation of the control data value supplied to the voice.

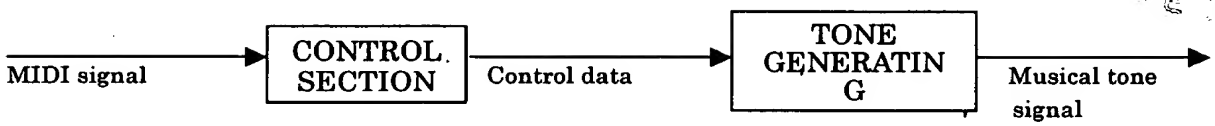
FIG. 8 shows a typical envelope of a musical tone.

[Description of reference numerals]

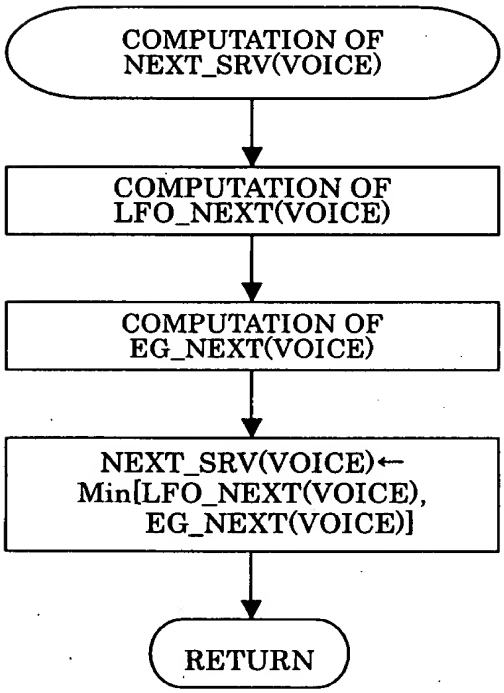
1.... Control section

2....Musical tone generating circuit

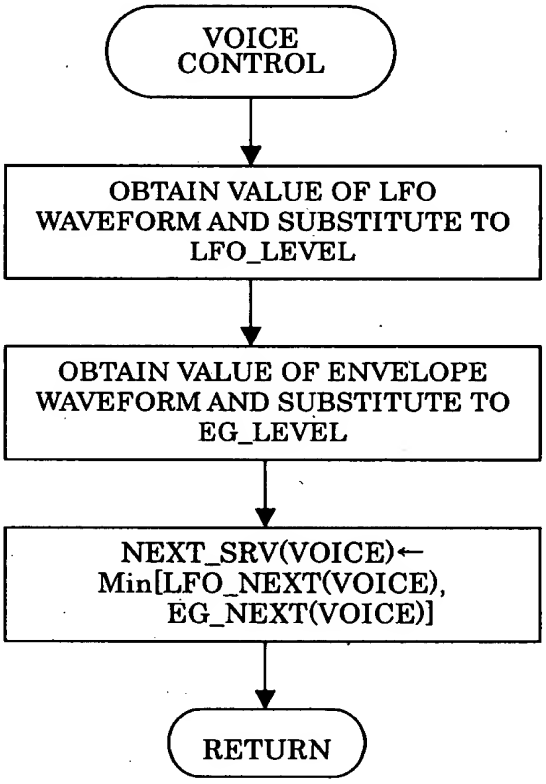
【Fig. 1】



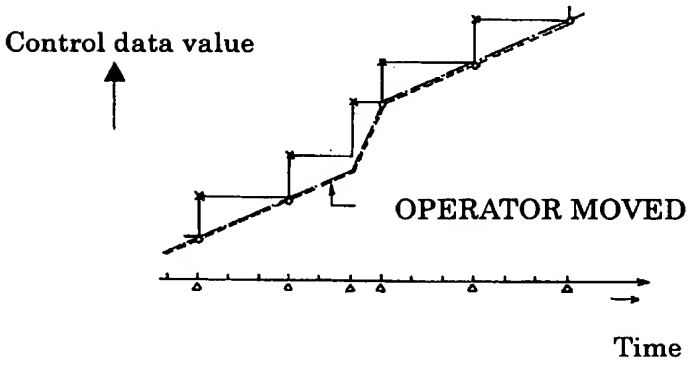
【Fig. 4】



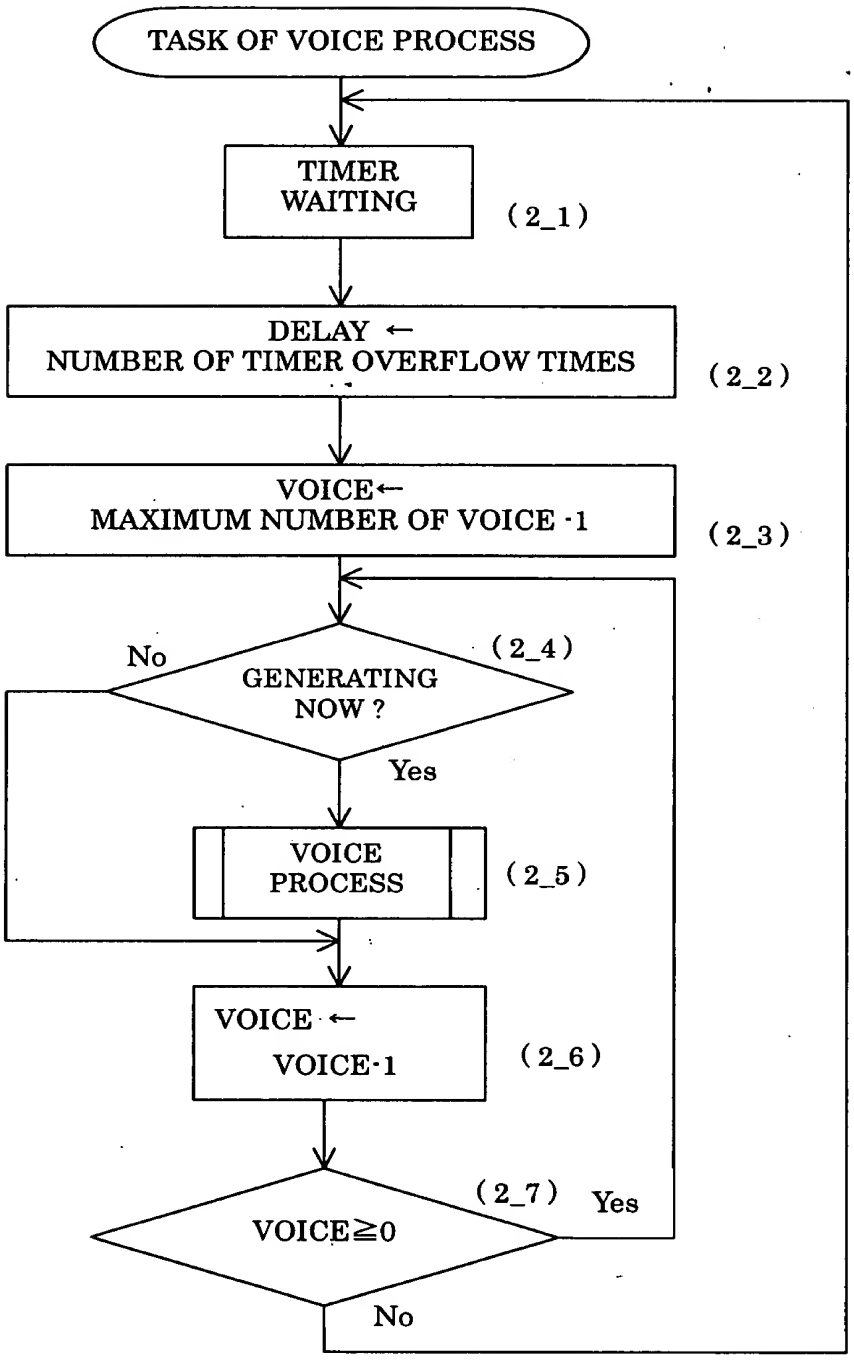
【Fig. 5】



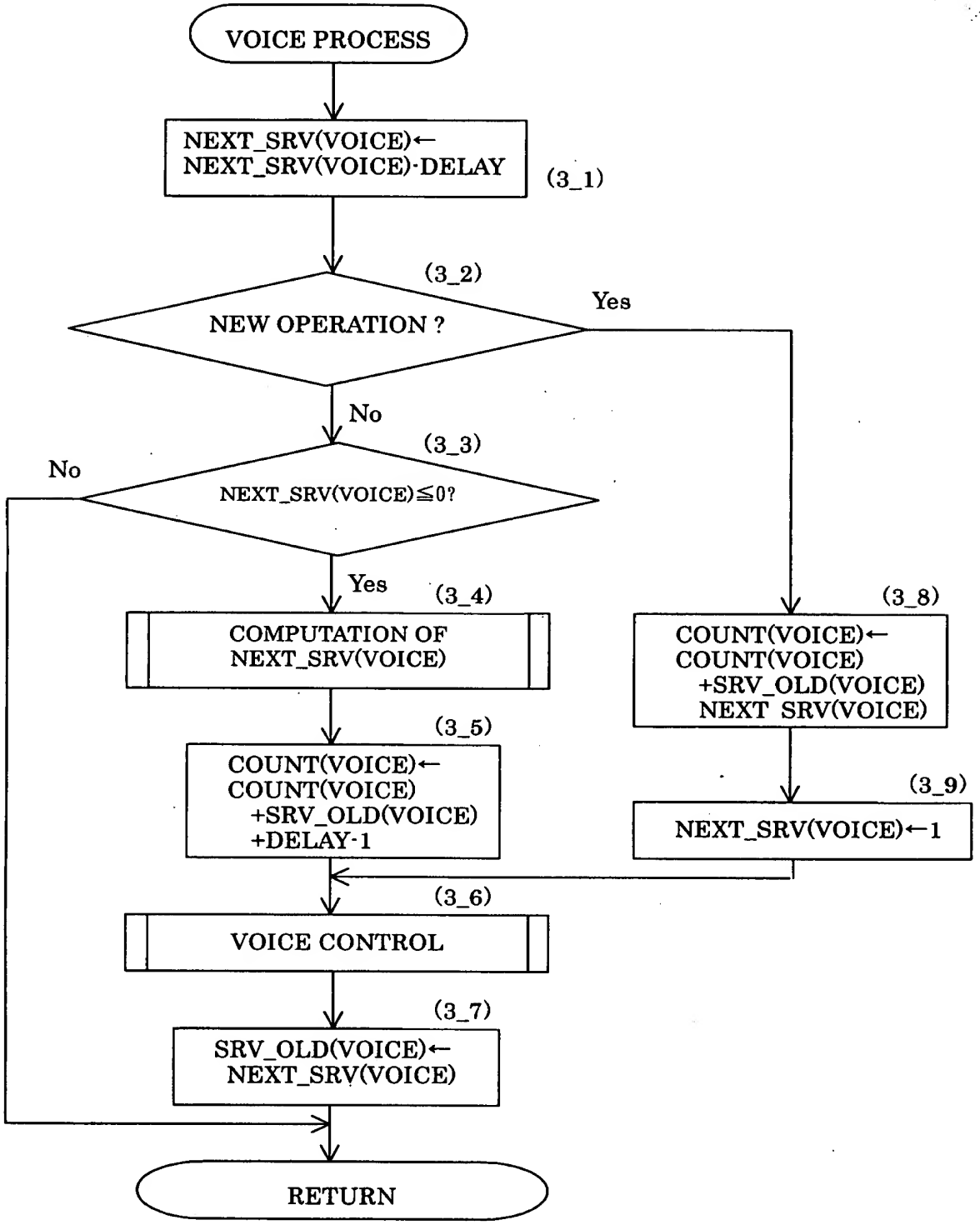
【Fig. 7】



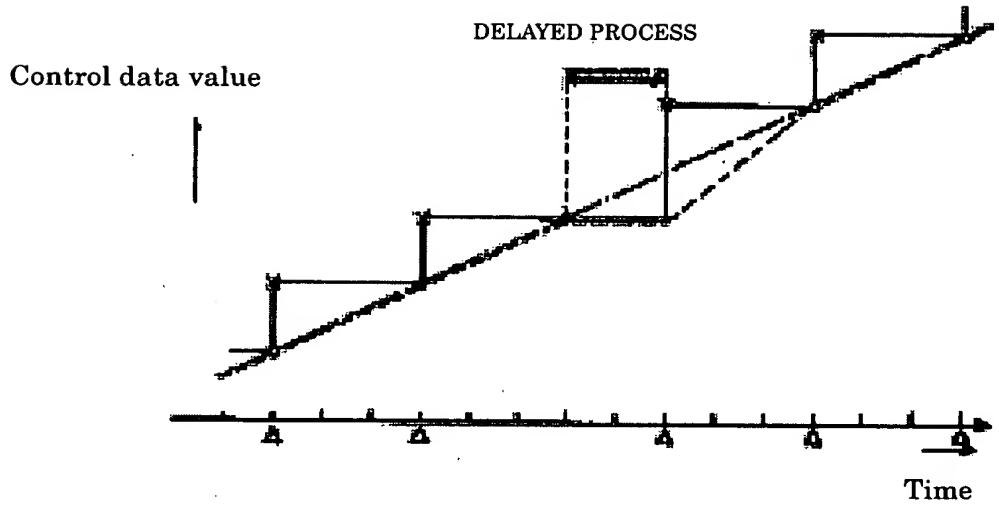
【Fig. 2】


TASK OF VOICE PROCESS
TIMER WAITING
(2_1)
DELAY ← NUMBER OF TIMER OVERFLOW TIMES
(2_2)
VOICE← MAXIMUM NUMBER OF VOICE -1
(2_3)
No
GENERATING NOW ?
(2_4)
Yes
VOICE PROCESS
(2_5)
VOICE ← VOICE-1
(2_6)
VOICE≧0
(2_7)
Yes
No

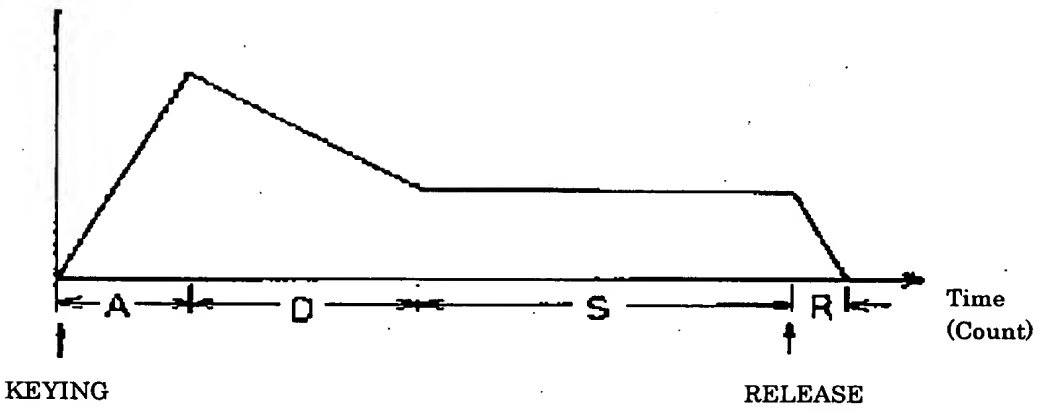
【Fig. 3】


VOICE PROCESS
NEXT_SRV(VOICE)←
NEXT_SRV(VOICE)-DELAY
(3_1)
(3_2)
NEW OPERATION ?
Yes
No
(3_3)
No
NEXT_SRV(VOICE)≦0?
Yes
(3_4)
COMPUTATION OF
NEXT_SRV(VOICE)
(3_5)
COUNT(VOICE)←
COUNT(VOICE)
+SRV_OLD(VOICE)
+DELAY-1
(3_8)
COUNT(VOICE)←
COUNT(VOICE)
+SRV_OLD(VOICE)
NEXT SRV(VOICE)
(3_9)
NEXT_SRV(VOICE)←1
(3_6)
VOICE CONTROL
(3_7)
SRV_OLD(VOICE)←
NEXT_SRV(VOICE)
RETURN

【Fig. 6】


DELAYED PROCESS
Control data value
Time

【Fig. 8】


A
D
S
R
Time
(Count)
KEYING
RELEASE